

Einführung in die Medieninformatik

Für EMI im 1. Semester beim Bart

Allgemeines

Aufgabenstellungen

Berechenbarkeit (Lösen mathematischer Probleme), Unberechenbarkeit, Churchsche These (Gleichwertigkeit aller Maschinenkonzepte), Chomsky-Hierarchie (Zusammenhang Automaten & Sprachgrammatik), Programmkomplexität, Programmverifikation

Medienkompetenz

Medialitätsbewusstsein + Medienwissen, Medienspezifisches Rezeptionsmuster, Medienbezogene Genussfähigkeit (Bedürfnis nach Identifikation und Unterhaltung), Medienbezogene Kritikfähigkeit.
Medien: Mittel zur Speicherung von Information (Physisch oder Format, Text, Video, Audio,)

Text und HTML

Zeichensätze

Zeichensatz definiert Zuordnung von Zahlen zu Zeichen.

1. **ASCII**: 7-bit-Zeichensatz (128 Zuordnungen), 0-31 Steuerzeichen. Neuere Zeichensätze sind meist abwärtskompatibel (Erste 128 Zuordnungen entsprechen ASCII)
2. **ISO-8859-1, ISO-8859-2 ... ISO-8859-16**: 8-bit-Zeichensätze (256 Zuordnungen), 0-127 ist ASCII, darüber Sonderzeichen wie deutsche Umlaute (-1: Latin1 westeuropäisch, -2: Latin2 osteuropäisch, -5: Kyrillisch, -6: Arabisch, -7: Griechisch [...], -15: Westeuropäisch mit Eurozeichen)
3. **Unicode**: Internationaler Zeichensatz. Verschiedene Standards: UCS-2 mit 2 hoch 16 Zeichen, UCS-4 mit doppelt so vielen Ebenen. Unicode 5 kennt 99k Zeichen.
Unicode-Formate:

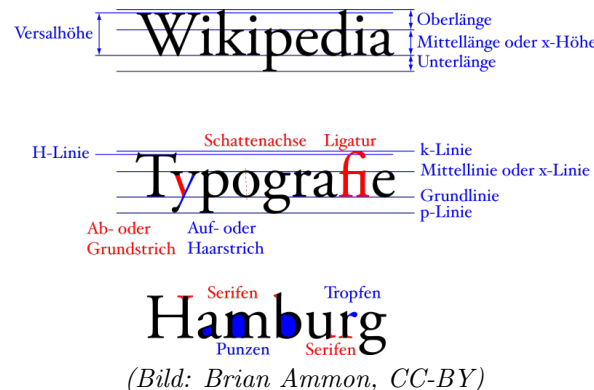
- **UTF-32**: Immer 4 Bytes für ein Zeichen
- **UTF-16**: 2 Bytes pro Zeichen mit Möglichkeit für mehr, wenn in ersten 2 Bytes spezielle Surrogate-Kodierung benutzt wird
- **UTF-8**: Wichtigstes Unicode-Format, weit verbreitet z. B. für XML. Besteht aus einem Byte pro Zeichen, das nur ASCII ist, wenn das oberste Bit gesetzt ist, ansonsten gehören die nächsten Bytes auch mit zum Zeichen, dann 2-4 Bytes. Aufbauschema: Höchstes Bit ist nicht gesetzt, wenn es nur ein ASCII-Zeichen ist und somit nur aus einem Byte besteht. Ist das höchste Bit gesetzt, gibt die Anzahl der dann folgenden Bits die 1 sind an, aus wie vielen Bytes das Zeichen besteht. Ist das erste Byte z. B. 11100000, folgen noch zwei weitere Bytes für das gleiche Zeichen.

Braille-Schrift

Von Louis Braille 1820 für Blinde entwickelt, für viele natürliche Sprachen verfügbar. Gibt Zeichen als fühlbare Punkte auf dem Papier wieder (6 - 8 Punkte kodieren ein Zeichen). Spezielle Schreibweisen für Formeln, Musik o. ä.

Schrift

Eigenschaften



1. Höhe wird in Punkt angegeben. Ein Punkt ist

meist 1/72 Zoll (TeX, DTP / Postscript), 1 Pica sind 12 Punkte.

2. Schriftstärke (mager, normal, fett), Schriftbreite (schmal, normal, breit), Schriftlage (normal, kursiv)
3. Vektorschriften: Definition durch Vektoren, Rasterschriften: Definition durch Pixelgrafik für je ein Zeichen
4. Serifen: Helfen bei Lesbarkeit, sollte nur bei Fließtext benutzt werden.
5. Kerning: Anpassung von Zeichenabständen bei parallelen Diagonalen ("BRAVO": Zusammenrücken von A und V), Definition pro Buchstabenpaar. Kann doof sein bei geringen Abständen.

Absätze

Absatzformat gibt Abstände und Einrückungen an. Möglich sind z. B. Blocksatz (Automatische Anpassung des Wortabstandes, sodass stets ganze Zeile gefüllt wird), Flattersatz / Mittelachsensatz (optimale Wortabstände). Seitenlayout: Kopf, Rumpf, Fuß.

Formate

Eigenschaften: Editierbar? WYSIWYG? Lesbar? Medien einbindbar? Funktionsumfang? Programmierbar?

• Plain

- **Rich Text** (Microsoft), eigene Definition für Medien, nicht programmierbar, alt, WYSIWYG, MS Word, LaTeX, PostScript, PDF, XHTML -

```
{\rtf Hallo \par {\i Dies} war kursiv. \par Noch ne Zeile.}
```

- **LaTeX** (Frei), voll programmierbar, kein WYSIWYG (wirft PDF o. ä. aus), Medien einbindbar, gute Formelsetzungsbibliotheken, vielseitig und toll, aber kompliziert
- **PDF** (Adobe), baut auf Postscript auf, Layout-Treu, Rechtemanagement, Einbetten von Schriften problematisch, kaum editierbar, wird aus anderen Formaten generiert

- **HTML:** Trennung von Struktur und Style mit CSS, Scripting (Document Object Model: Abbildung des XML-Baums auf JavaScript), programmierbar, kein WYSIWYG, Plugins

Internetz

- **HTTP:** Protokoll zwischen Web-Server und Web-Client (Browser)
- **URI:** Bezeichnung und Adressierung beliebiger Daten im Netz (z. B. `http://www.wh2.tu-dresden.de`)
- **Hypertext:** "Nicht-lineares" Textmedium, verbindet Inhalte durch Hyperlinks (URI, entfernte URIs wie "`http://google.de`", "`ftp://...`" oder lokale URIs wie "`mailto:`", die an andere Programme auf dem Rechner gehen)
- **Mark-Up-Sprachen:** Beschreibung von Text durch Text, bspw. HTML mit öffnenden Tags und schließenden Tags und Inhalt dazwischen

HTML

```
<html><head><title>Hallo Welt</title>
</head><body><h1>Grosse Ueberschrift
</h1><a href="uri">Link</a></body></html>
```

- Attribute bspw. href mit Ziel-URI, align mit Ausrichtung o. ä.
- `<a>`-Elemente sind Anker, verlinkt wird per "`#ziel`" und gesetzt per name-Attribut
- Bilder mit ``
- Listen mit `ul` (unordered list) und `ol` (ordered list)-Elementen, Einträge mit `li` (list item)-Elementen
- Tabellen mit `table`, `tr` für Zeilen und `td` für Spalten.

```
<table><tr><th>1</th><th>2</th></tr>
<td>A</td><td>B</td></tr></table>
```

- `div` für allgemeine Blöcke

Barrierefreiheit

Barrieren für Behinderte auf Webseiten (Sehbehinderte, Farbenblinde, Gehörbehinderte, Motorisch behinderte, geistig behinderte)

CSS

Strikte Trennung zwischen Präsentation und Design, definierte Formate für XHTML-Tags. Text-Formatierung, Positionierung von Elementen, Rahmen und Leerräume, Hintergrund, Formatierung von Tabellen, CSS einbinden mit `<style type="text/css"></style>` für direktes reinschreiben von CSS, CSS-Datei im Head des HTML-Dokuments verlinken mittels `<link rel="stylesheet" type="text/css" href="datei.css">`. HTML-Elemente koennen auch mit style-Attribut und CSS-Angaben gestyled werden.

CSS besteht aus Selektoren für Elemente (mittels ID oder Klasse) und Style-Attributen. Bspw.: `#foo { color: red; }` setzt die Schriftfarbe im Element mit der ID foo auf rot. Mögliche Styleattribute sind z. B. "`font-family`" (Arial, sans-serif), "`font-weight`" (bold).

Positionierung

Mit CSS werden Elemente nach dem Boxmodel layoutet, d. h. jedes Element bildet eine Box. CSS-Attribute für jedes Element sind z. B. "`margin`" (Außenrand um die Box des Elements, z. B. "`5px`"), "`padding`" (Innenabstand um den Boxrand), "`border-width`" (Randbreite), "`border-color`" (Randfarbe).

Navigationstechniken

Websites sollten sets einwandfrei navigierbar sein (vor / zurück, Nutzer muss sich zurechtfinden können). Bspw. durch Setzen von Navigationsleisten, Suchmaschinen etc.

Karten in HTML

Map-Element definiert Karte. Bsp.:

```
<map name="schland">
<area shape="rect" coords="11,10,59,29"
href="uri" alt="Hier"></map>

```

Beispielsweise rect mit `coords="x1,y1,x2,y2"` mit 1=Obere Linke Ecke, 2=Untere rechte Ecke oder "`circle`" mit "`x,y,r`" oder "`poly`" "`x1,y1,x2,y2...`"

Dynamisches Web

Dynamik z. B. durch auf dem Web-Server dynamisch generierte Inhalte durch PHP, ASP.net o. ä. Im Browser wird JavaScript benutzt.

JavaScript

1. **Objekt:** Ein JavaScript-Objekt besteht aus Schlüssel, denen Werte zugeordnet werden (bspw. das document-Objekt hat den Schlüssel bgcolor mit dem Wert white).
2. **Funktionen / Methoden / Aktionen:** Wie Funktionen in C, können auch Variablen zugeordnet werden
3. **Ereignisse:** Das Ereignis "onmouseover" eines HTML-Elements kann durch JavaScript auf eine Funktion gesetzt werden.
4. **Variablen:** Können Zahlen, Strings, Booleans, Objekte, Funktionen sein. Bsp. zur Deklaration: `var month = "Januar";`
5. Ausdrücke ordnen einer Variable einen Wert zu (Beispiel s. o.)
6. Operatoren verändern Variablen (siehe C, bspw. "+", "=", ...)